

CLAIMS

What is claimed is:

1 1. A system for performing interleaved packet processing
2 in a network router, wherein a packet to be routed includes
3 a source address bit pattern and a destination address bit
4 pattern that are each processed by a task processor in
5 accordance with a data tree, said data tree including a
6 plurality of nodes linked by branches wherein an
7 instruction that is associated with each node within said
8 data tree is utilized for determining which branch is to be
9 taken in accordance with said source address bit pattern or
10 said destination address bit pattern, said system
11 comprising:

12 a first bank of registers for loading an instruction
13 to be executed by said task processor at each node of said
14 data tree in accordance with said source address bit
15 pattern;

16 a second bank of registers for loading an instruction
17 to be executed by said task processor at each node of said
18 data tree in accordance with said destination address bit
19 pattern; and

20 a task scheduler for enabling said first bank of
21 registers to transfer an instruction loaded therein for
22 processing by said task processor only during even time
23 cycles and for enabling said second bank of registers to
24 transfer an instruction loaded therein for processing by
25 said task processor only during odd time cycles.

2. The system of claim 1, wherein said task scheduler includes a clock signal generator that generates said even and odd time cycles in an alternating series of rising edges and falling edges.

3. The system of claim 1, further comprising an address register for storing an address of a next instruction to be loaded into either said first bank of registers or said second bank of registers from a memory device before being executed by said task processor.

4. The system of claim 3, wherein said address register further comprises a counter for incrementing said address of the next instruction in response to a dual instruction.

5. The system of claim 3, wherein said memory includes instructions to be executed by said task processor.

6. The system of claim 5, wherein said memory further comprises a first memory area containing normal size instructions and a second memory area containing dual size instructions.

7. The system of claim 1, further comprising at least one temporary register for storing information from said task processor between two consecutive processing time cycles when such a processing lasts more than one time cycle.

8. The system of claim 8, further comprising a 1-bit state register for each of said first and second bank of registers, said 1-bit state register being set when said processing lasts more than one time cycle.

1 9. A method for performing interleaved packet processing
2 in a network router, wherein a packet to be routed includes
3 a source address bit pattern and a destination address bit
4 pattern that are each processed by a task processor in
5 accordance with a data tree, said data tree including a
6 plurality of nodes linked by branches wherein an
7 instruction that is associated with each node within said
8 data tree is utilized for determining which branch is to be
9 taken in accordance with said source address bit pattern or
10 said destination address bit pattern, said method
11 comprising:

12 loading into a first bank of registers an instruction
13 to be executed by said task processor at each node of said
14 data tree in accordance with said source address bit
15 pattern;

16 loading into a second bank of registers an instruction
17 to be executed by said task processor at each node of said
18 data tree in accordance with said destination address bit
19 pattern;

20 transferring an instruction from said first bank of
21 registers to be processed by a task processor only during
22 even time cycles; and

23 transferring an instruction from said second bank of
24 registers to be processed by said task processor only
25 during odd time cycles.

1 10. The method of claim 9, further comprising generating
2 said even and odd time cycles in an alternating series of
3 rising edges and falling edges.

1 11. The method of claim 9, further comprising storing an
2 address of a next instruction to be loaded into either said
3 first bank of registers or said second bank of registers
4 from a memory device before being executed by said task
5 processor.

1 12. The method of claim 11, further comprising:

2 loading said address of said next instruction from
3 said task processor into said first or second bank of
4 registers;

5 transferring said address from said bank of registers
6 to said address register;

7 reading said address from said address register; and

8 fetching said next instruction from said memory in
9 response to said reading step.

10 13. The method of claim 11, further comprising
11 incrementing said address of the next instruction in
12 response to a dual instruction.

1 14. The method of claim 13, further comprising:

2 loading a dual instruction into either said first bank
3 of registers or said second bank of registers; and

4 interrupting said loading step during one time cycle
5 if said loading requires two time cycles; and

6 during said time cycle during which said loading is
7 interrupted, loading an instruction into the other of said
8 first or second bank of registers.

1 15. The method of claim 13, further comprising:

2 processing a dual instruction utilizing said task
3 processor;

4 interrupting said processing step during one time
5 cycle if such a processing requires two time cycles; and

6 during said time cycle during which said processing is
interrupted, executing an instruction provided by the other
of said first or second bank of registers.

7 16. The method of claim 9, further comprising storing
8 information from said task processor between two
9 consecutive processing time cycles when such a processing
10 lasts more than one time cycle within at least one
11 temporary register.